SNS COLLEGE OF TECHNOLOGY

Coimbatore-36.
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

COURSE NAME : 16CS307 PRINCIPLES OF COMPILER DESIGN

III YEAR/ VI SEMESTER

UNIT – II COMPILERS AND LEXICAL ANALYSIS

Topic: Regular Expression to NFA and NFA to DFA

Dr.B.Vinodhini
Associate Professor
Department of Computer Science and Engineering

Regular Expression to NFA

Upcoming Topics

1. Regular expression to NFA(Thompson Construction)
2. NFA to DFA(Subset Construction)
3. Direct conversion of Regular Expression to DFA

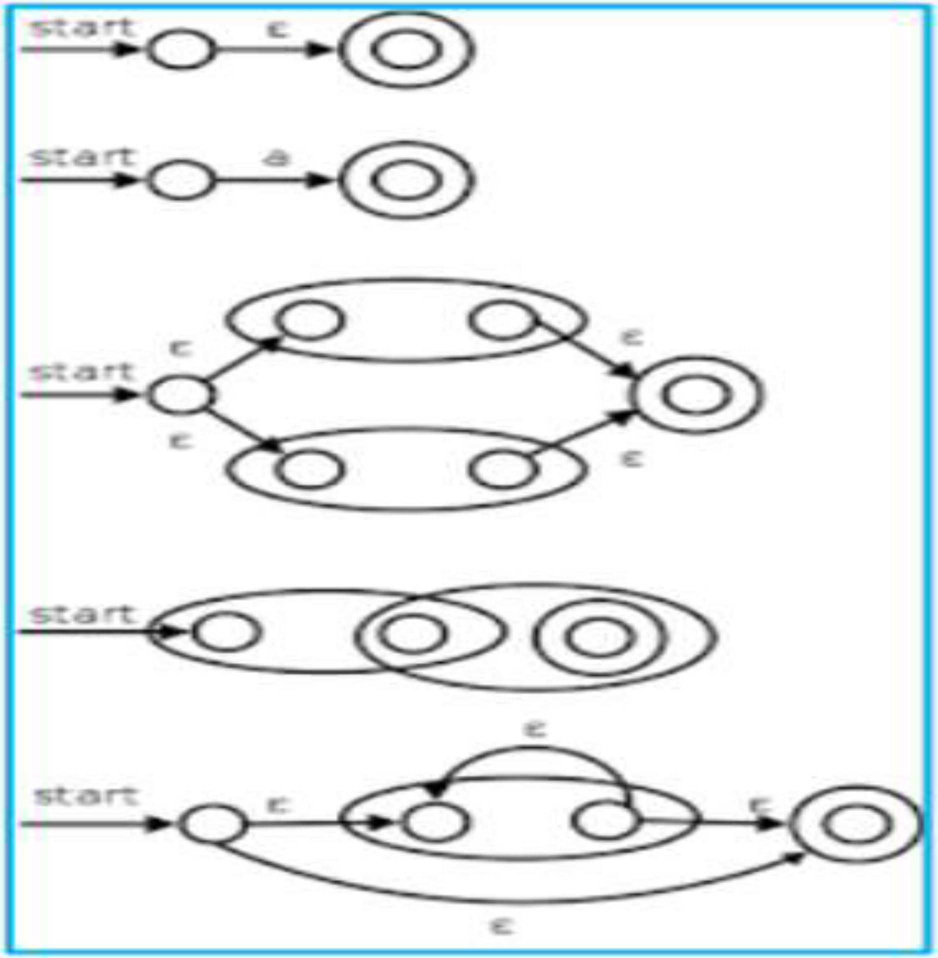
Regular Expression to NFA Thompson Construction Algorithm

Regular expressions over alphabet Σ

1. ε is a regular expression that denotes $\{\varepsilon\}$.

2. If **a** is a symbol (i.e., if **a** $\in \Sigma$), then **a** is a regular expression that denotes $\{a\}$.

3. Suppose r and s are regular expressions denoting the languages L(r) and L(s). Then
   a) (r) | (s) is a regular expression denoting L(r) U L(s).
   b) (r)(s) is a regular expression denoting L(r)L(s).
   c) $(r)^*$ is a regular expression denoting $(L(r))^*$.



Automatic lexical analyzer generation

- How do Lex and similar tools do their job?
  - Lex translates regular expressions into transition diagrams.
  - Then it translates the transition diagrams into C code to recognize tokens in the input stream.
- There are many possible algorithms.
- The simplest algorithm is RE -> NFA -> DFA -> C code.

Regular Expression to NFA

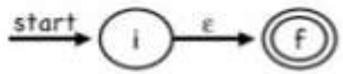
RE -> NFA algorithm (Thompson's Construction)

Inputs: A RE r over alphabet ∑

Outputs: A NFA N accepting L(r)

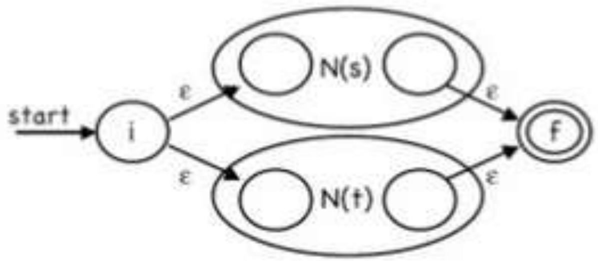
Method: Parse r.

If r = ε, then N is



If r = a ∈ ∑ , then N is

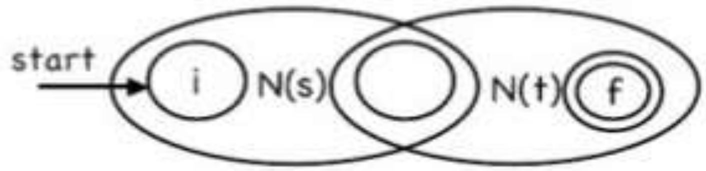
If r = s | t, construct N(s) for s and N(t) for t then N is



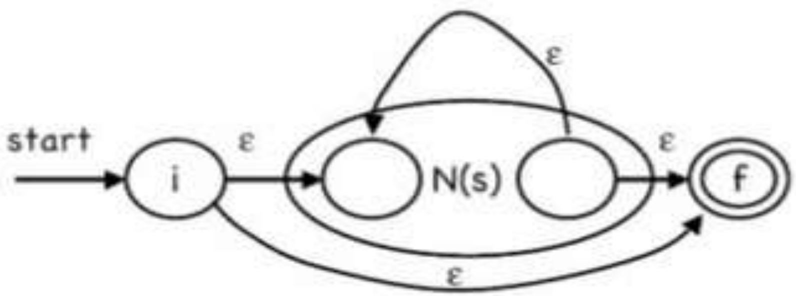
Regular Expression to NFA

RE -> NFA algorithm

If r = st, construct N(s) for s and N(t) for t then N is



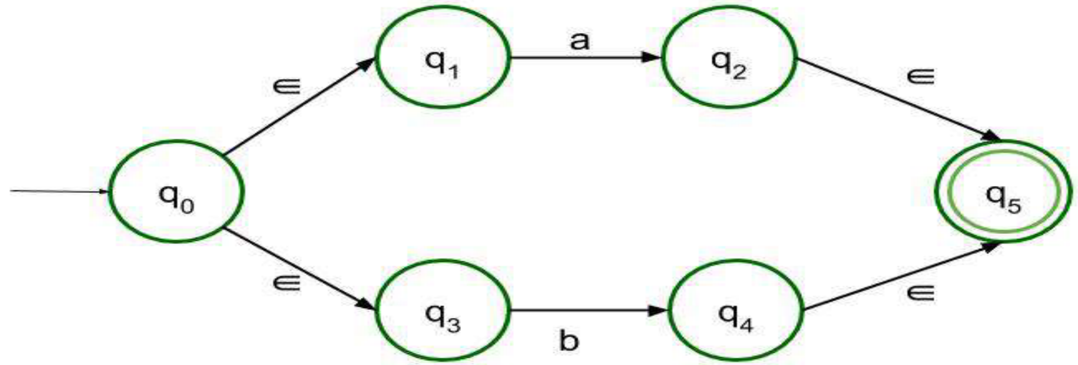
If r = s^*, construct N(s) for s, then N is



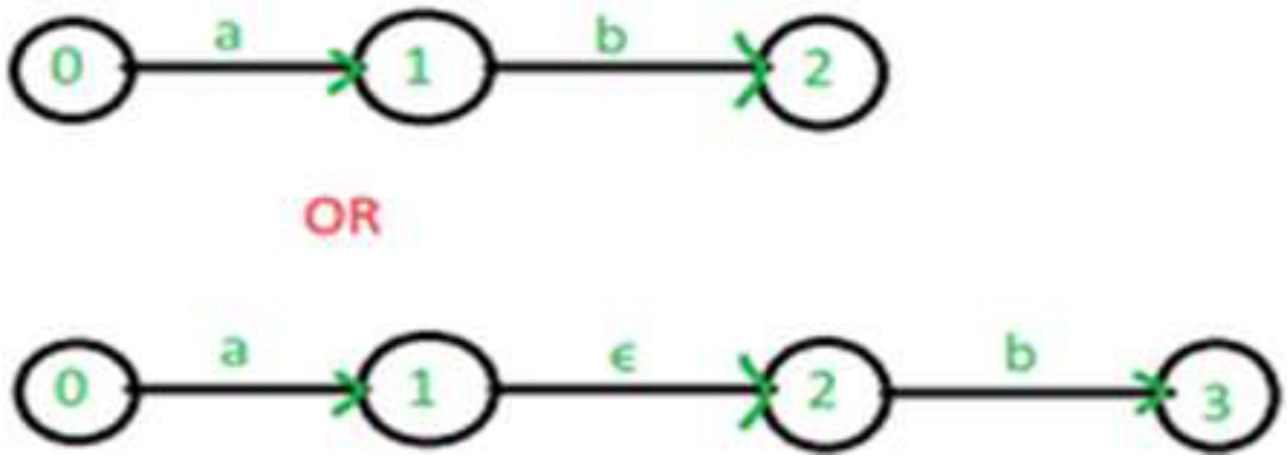
Regular expression to ε-NFA

Designing NFA using Thompson's Construction

- **(a+b)** OR **a|b**

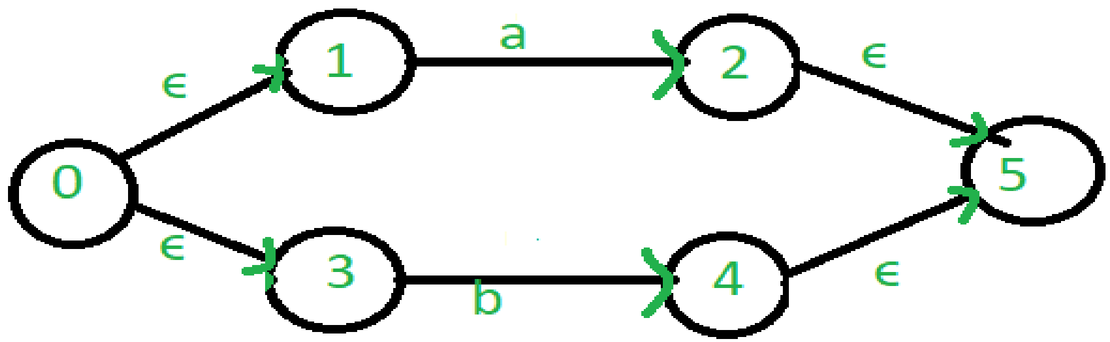


- **ab**

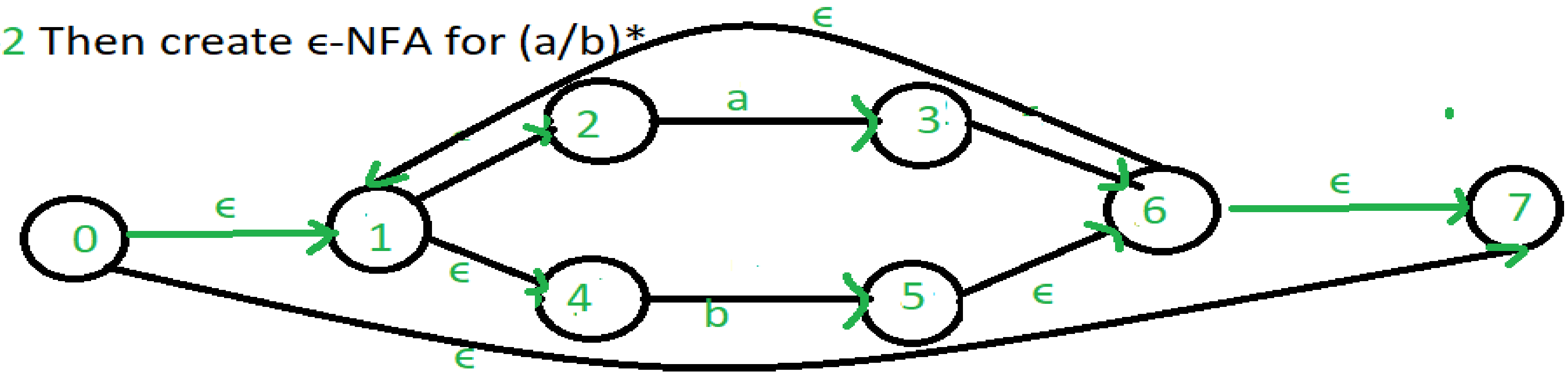


2. Regular expression to ε-NFA (a+b)*a

Step-1 First we create ϵ-NFA for (a/b)

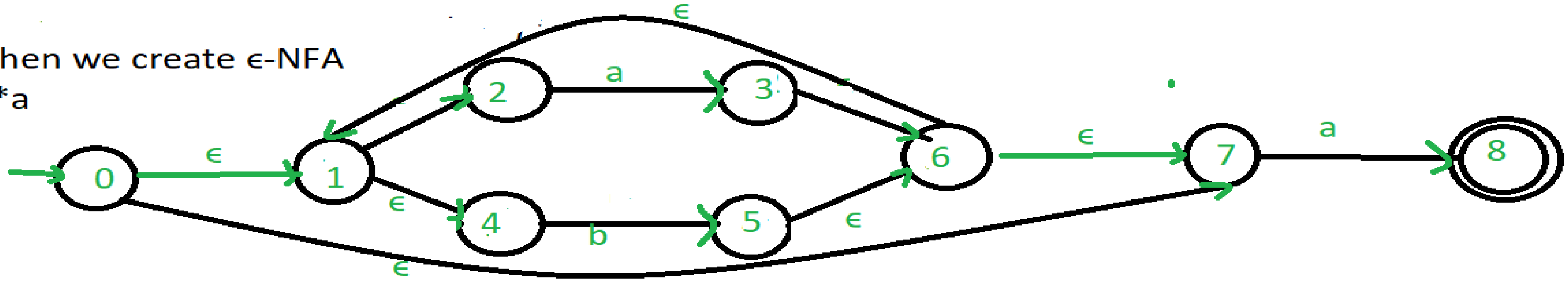


Step-2 Then create ϵ-NFA for (a/b)*



2. Regular expression to **ε-**NFA (a+b)*a

Step-3 Then we create ϵ-NFA for(a/b)*a

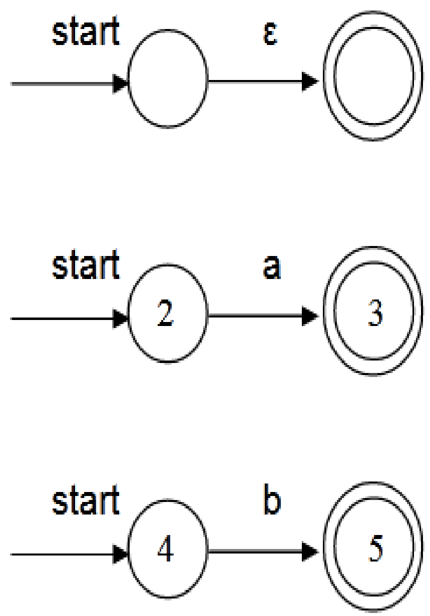


Regular Expression to NFA

Use the NFA Construction algorithm build the NFA for r (a|b)*abb

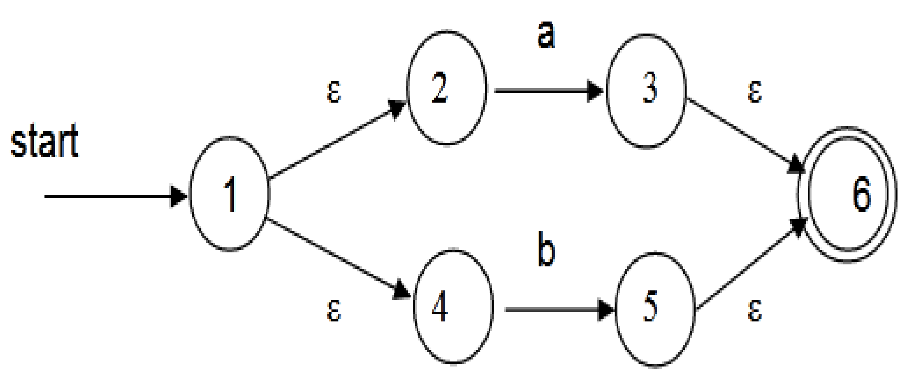
Take these NFA's in turn:

a. the NFA's for single character regular expressions ε, a, b



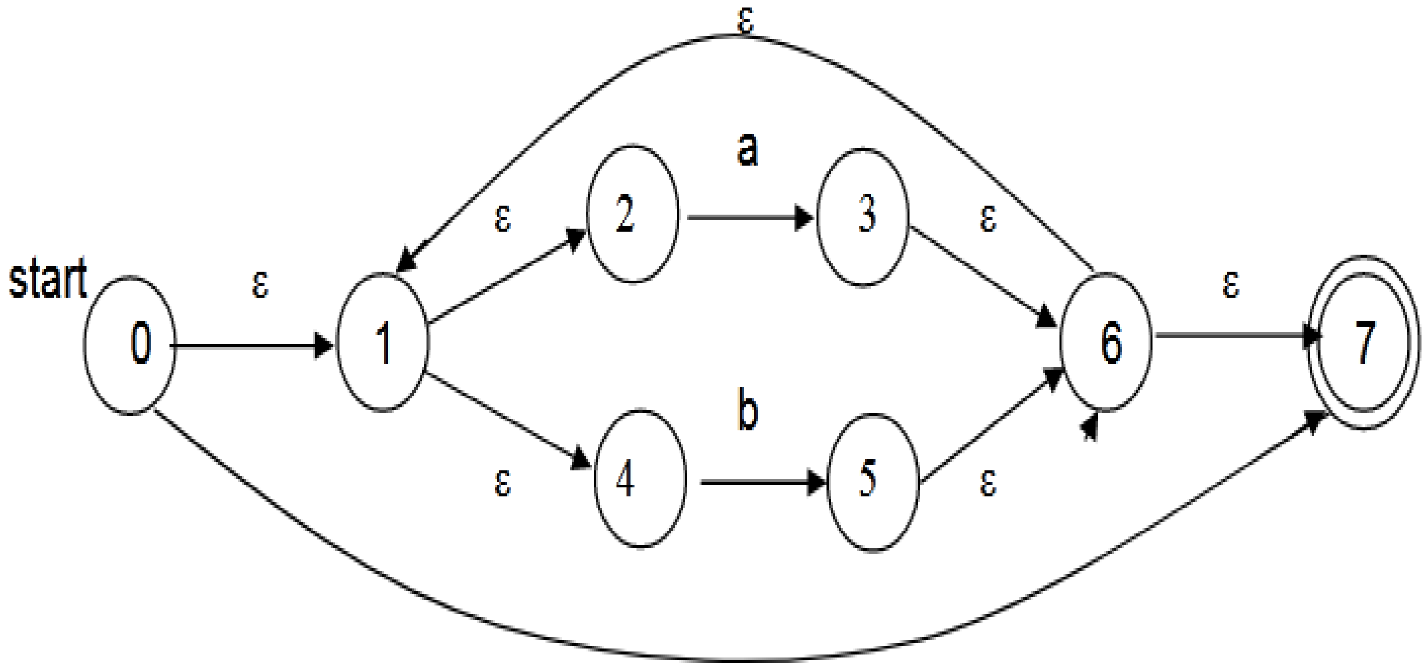
Regular Expression to NFA

b. the NFA for the union of a and b: a|b is constructed from the individual NFA's using the ε NFA as "glue". Remove the individual accepting states and replace with the overall accepting state



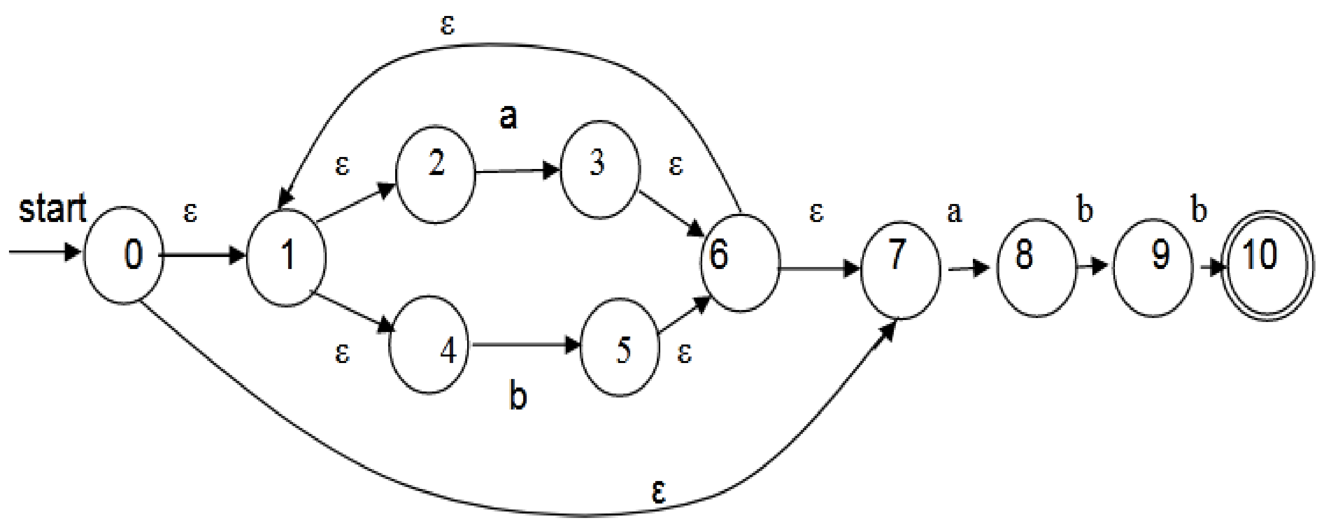
Regular Expression to NFA

c. Kleene star on $(a|b)^*$. The NFA accepts ε in addition to $(a|b)^*$



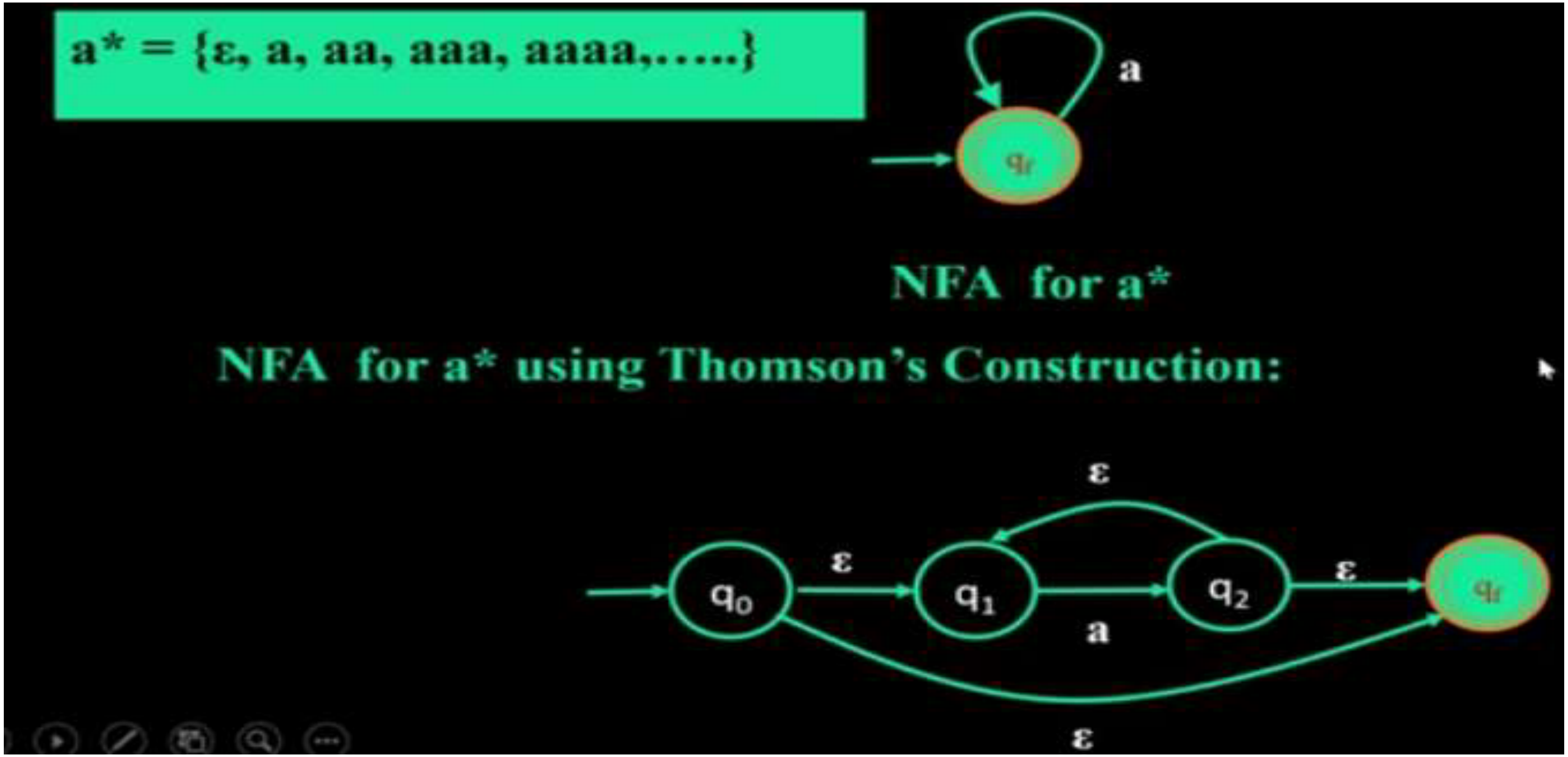
Regular Expression to NFA

d. concatenate with abb



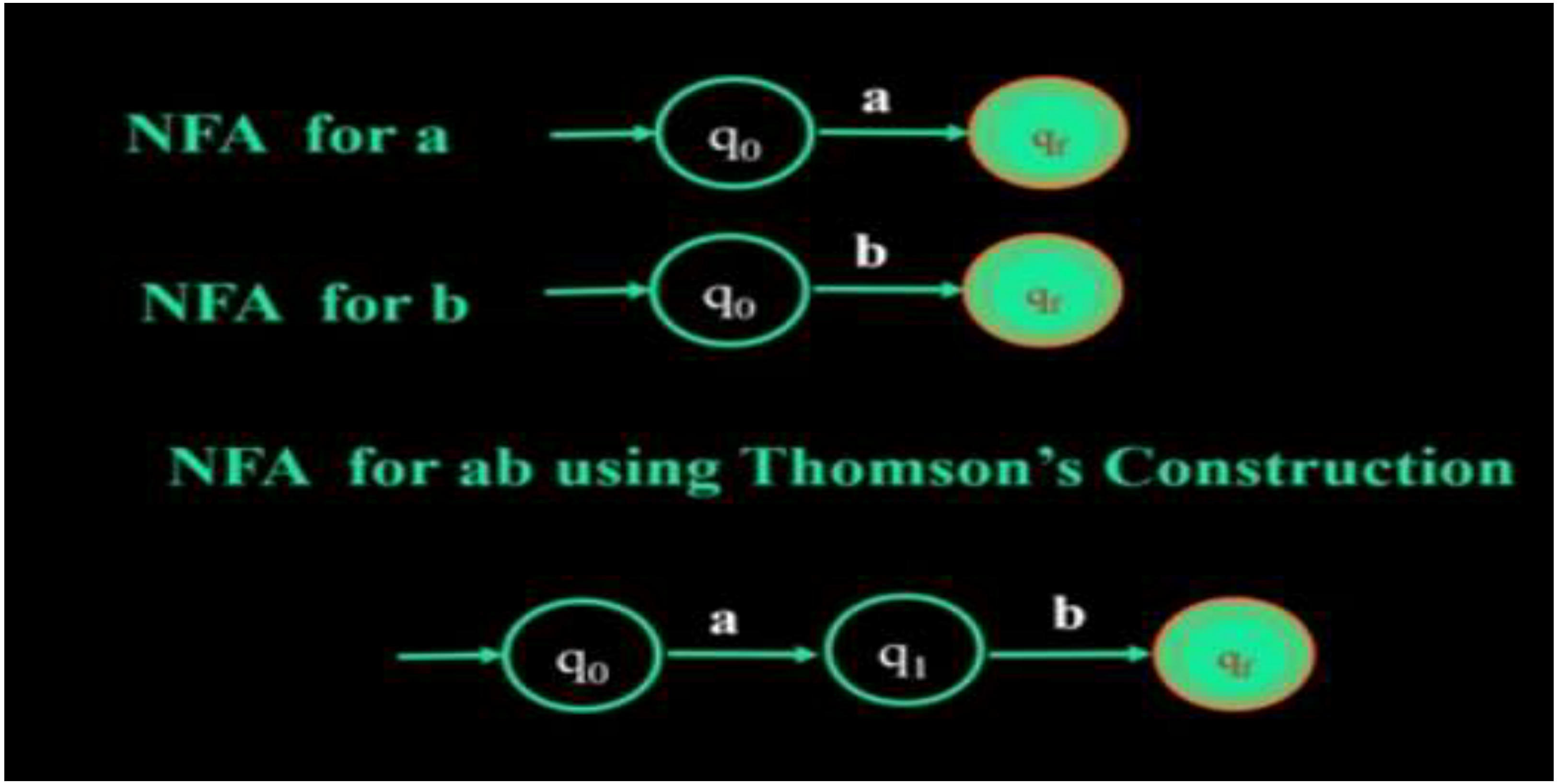
This is the complete NFA. It describes the regular expression (a|b)*abb. The problem is that it is not suitable as the basis of a DFA transition table since there are multiple ε edges leaving many states (0, 1, 6).

Regular Expression to NFA



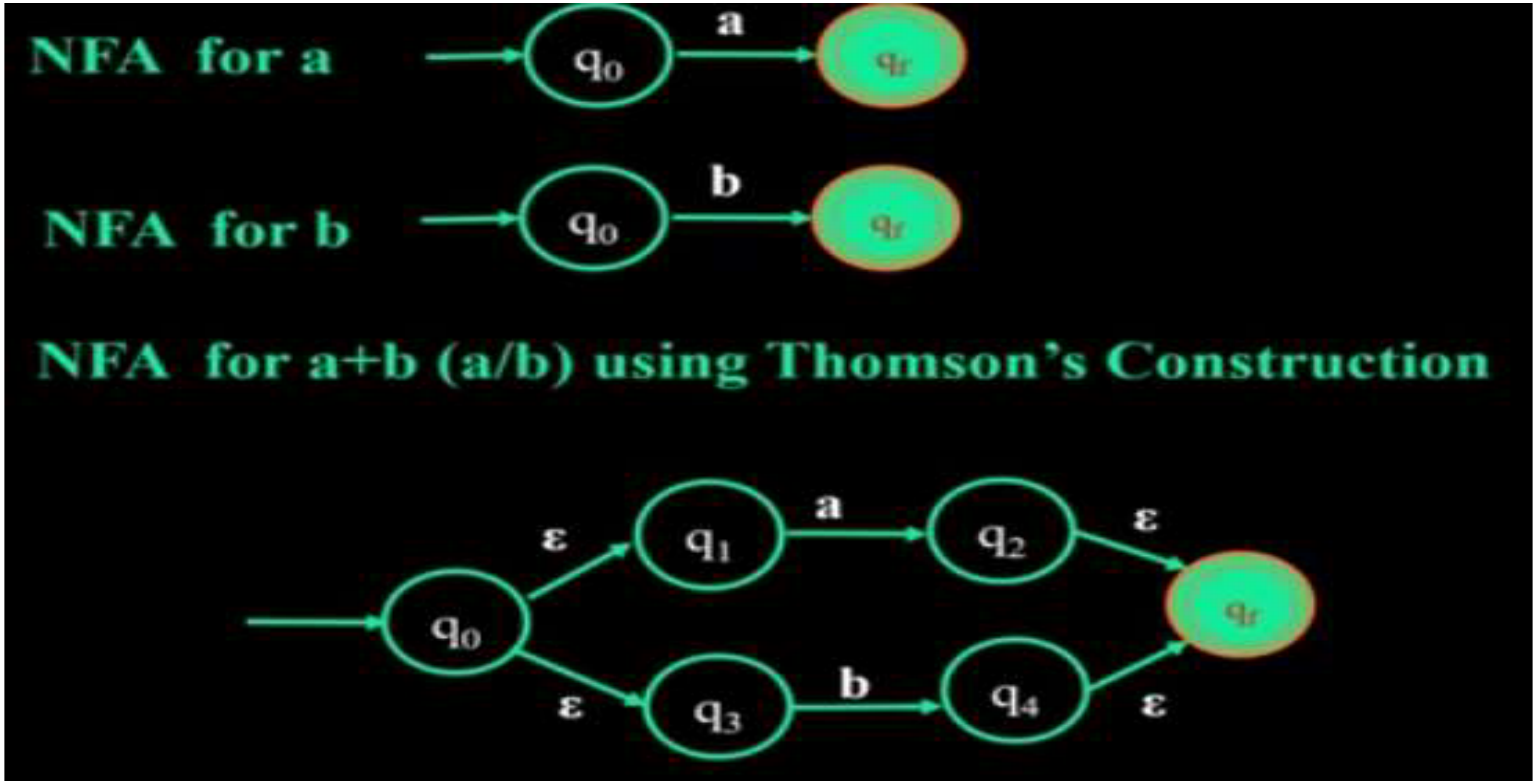
Regular Expression to NFA

Thompson Construction for Concatenation Operation

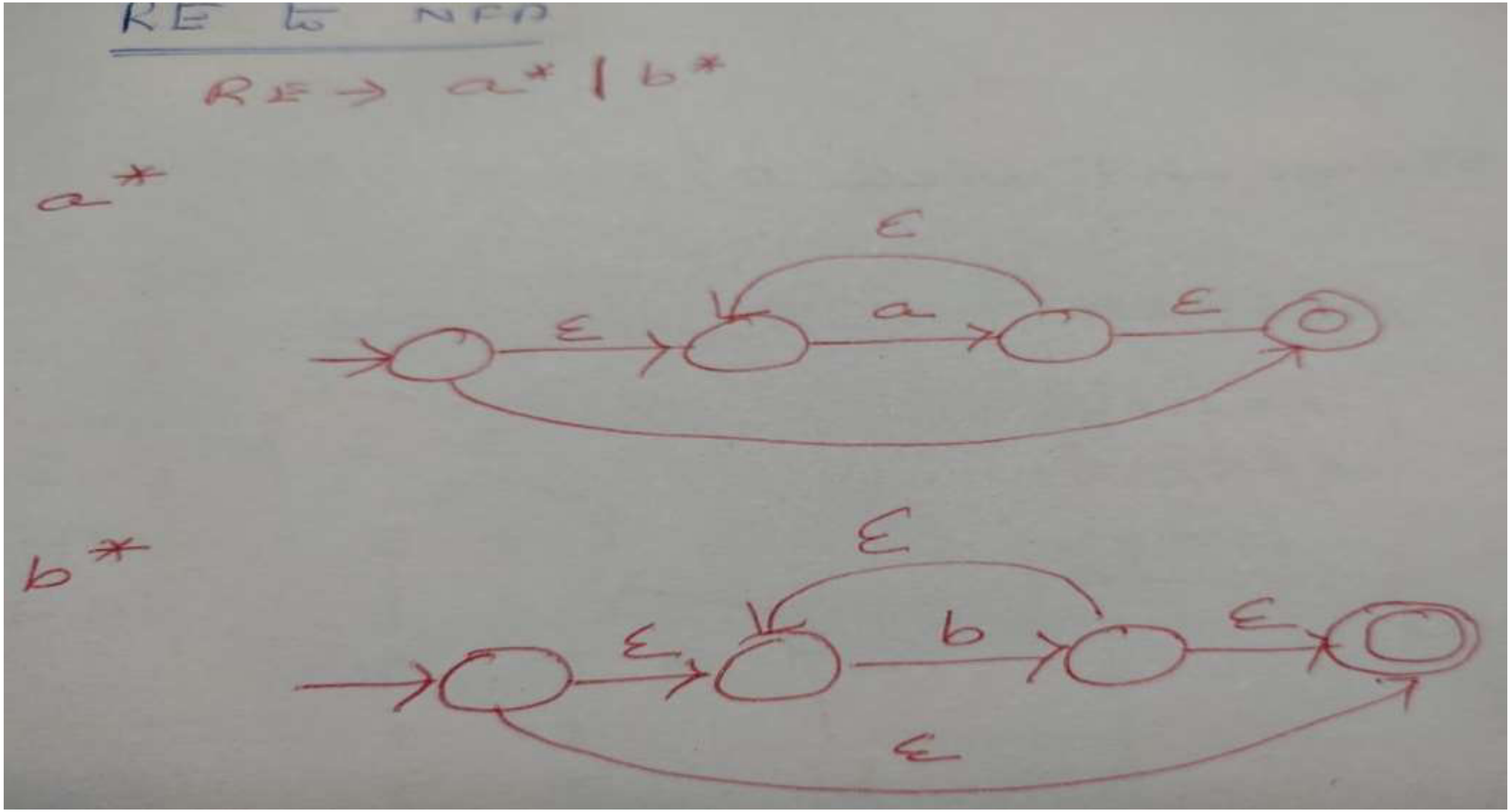


Regular Expression to NFA

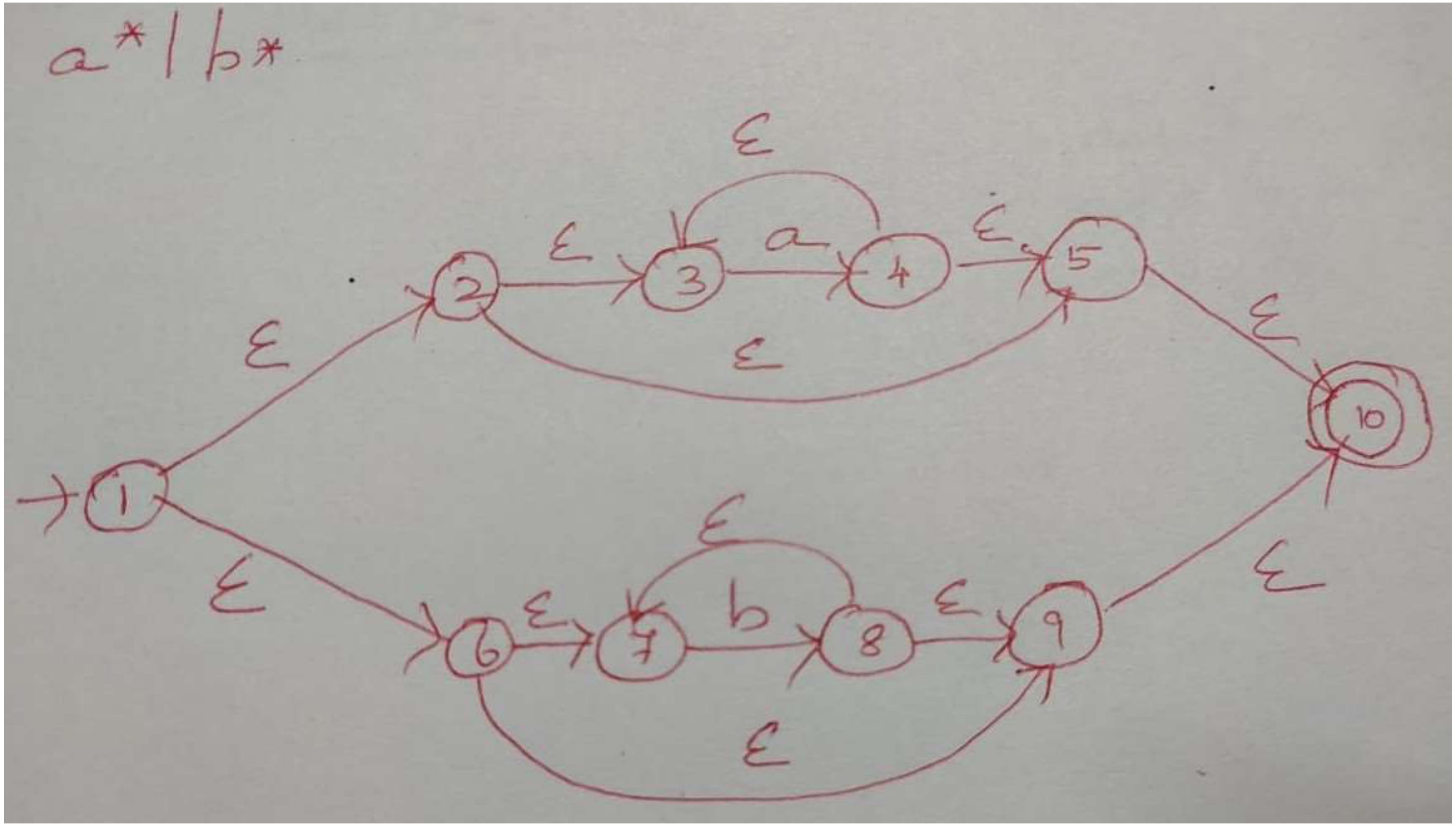
Thompson Construction for OR Operation



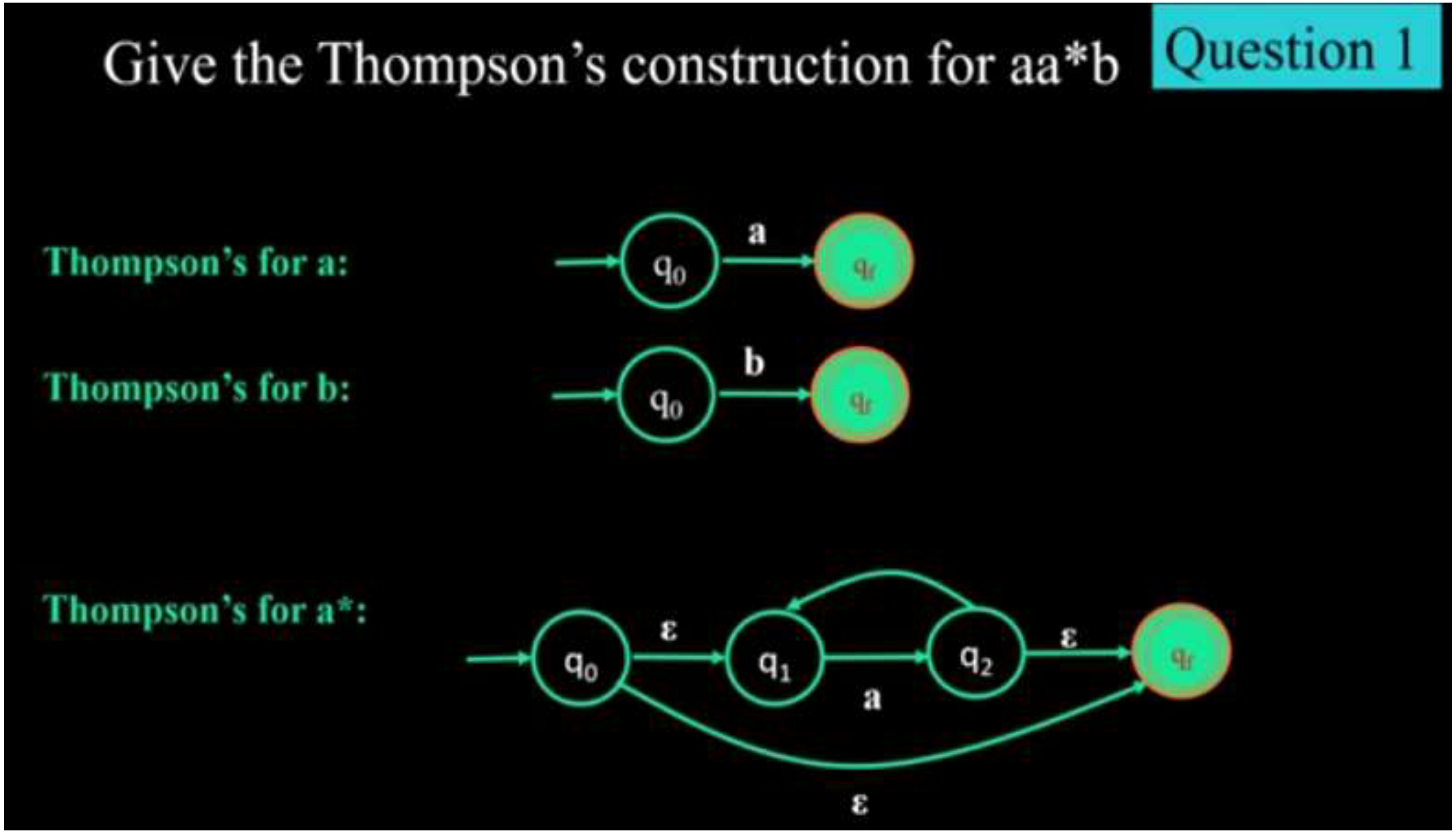
Regular Expression to NFA



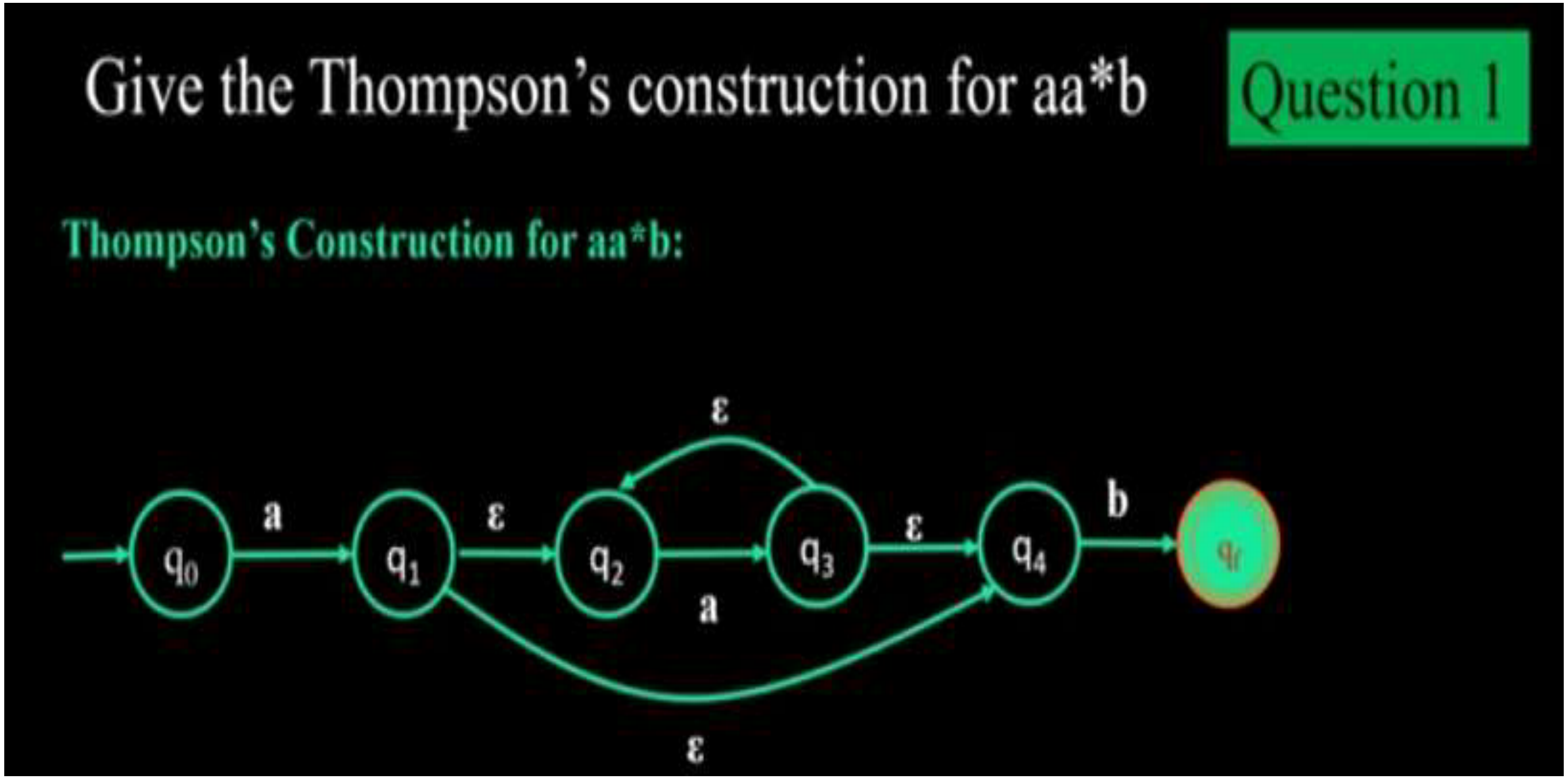
Regular Expression to NFA



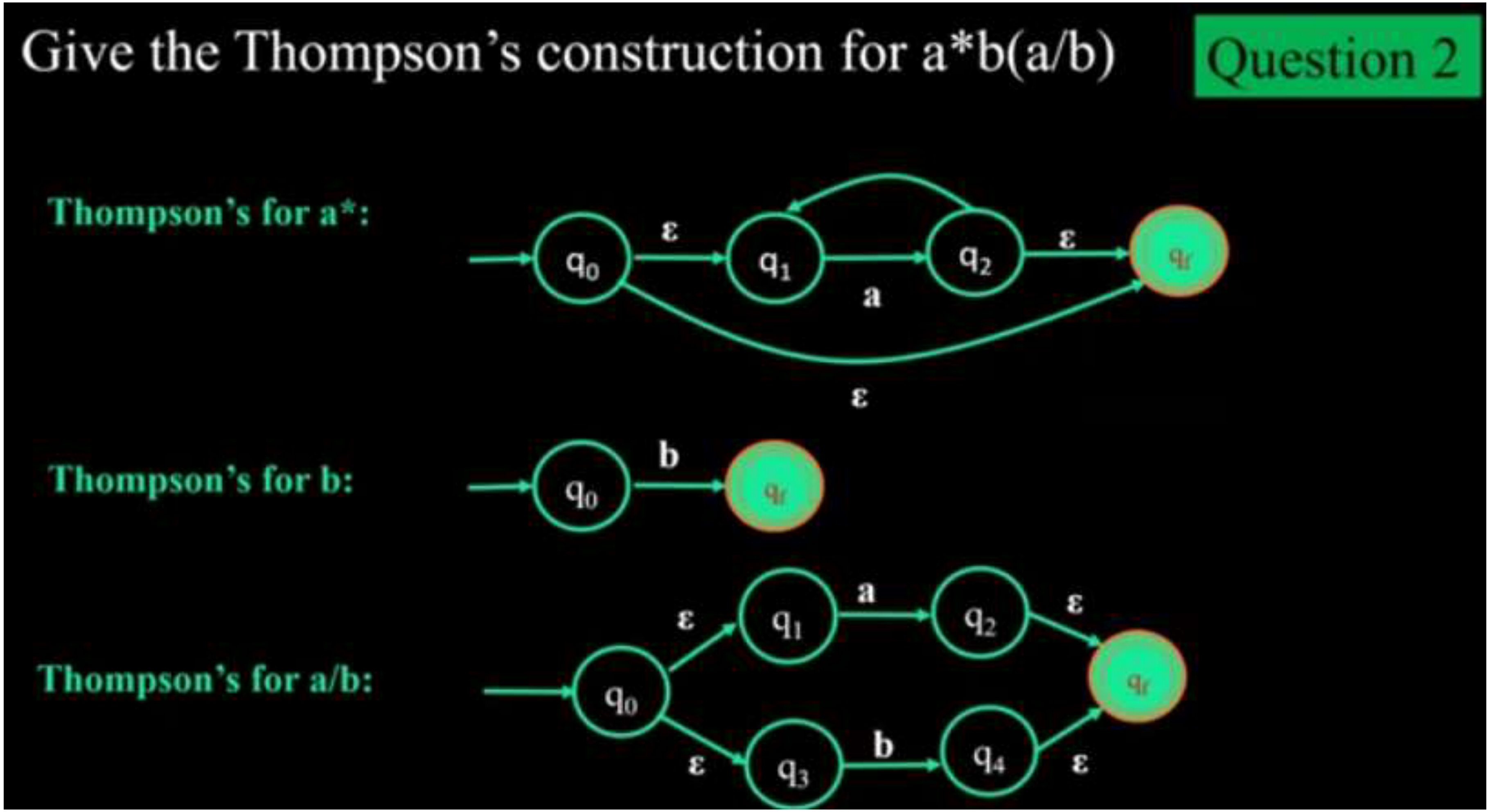
Regular Expression to NFA



Regular Expression to NFA



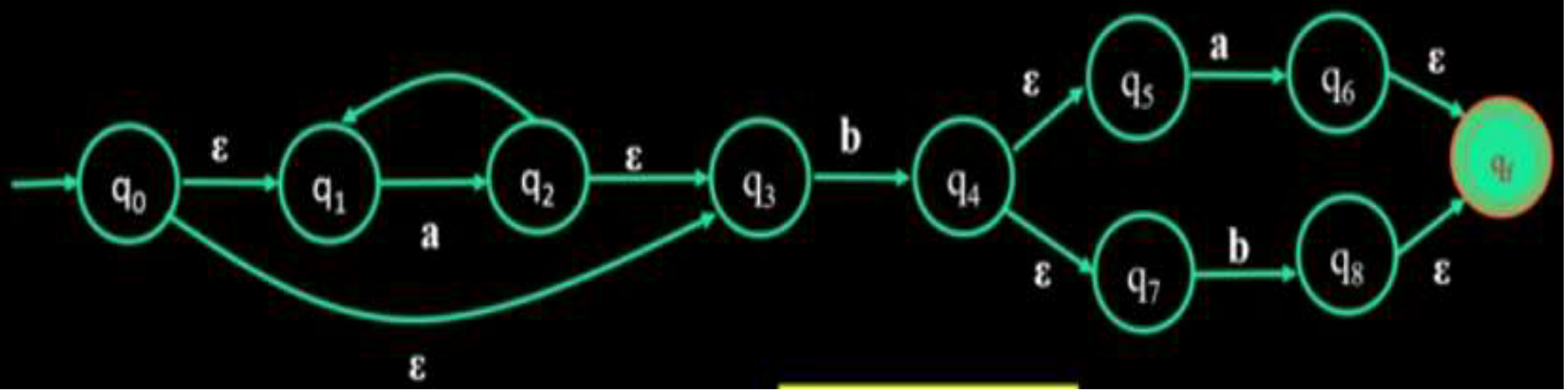
Regular Expression to NFA



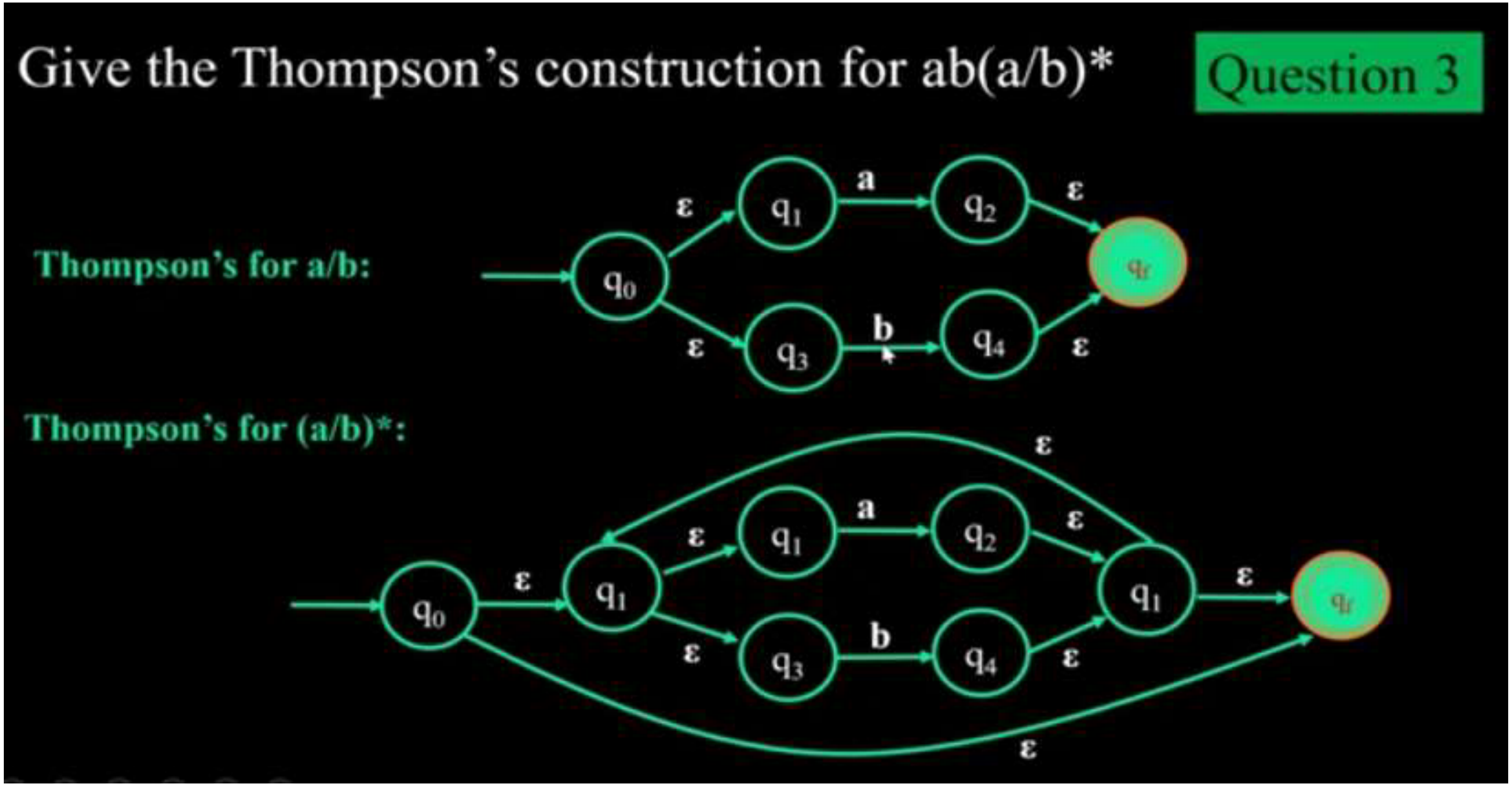
Regular Expression to NFA

Give the Thompson's construction for a*b(a/b)

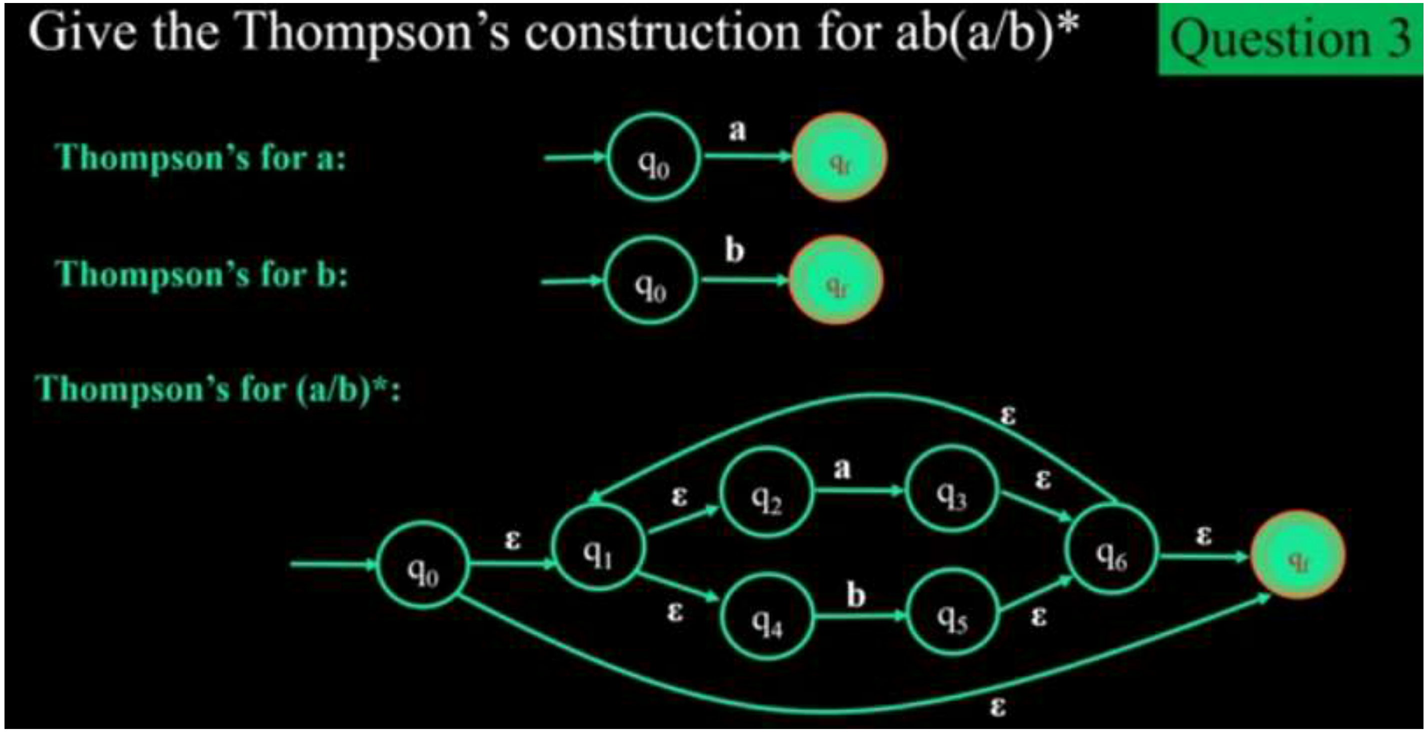
Question 2



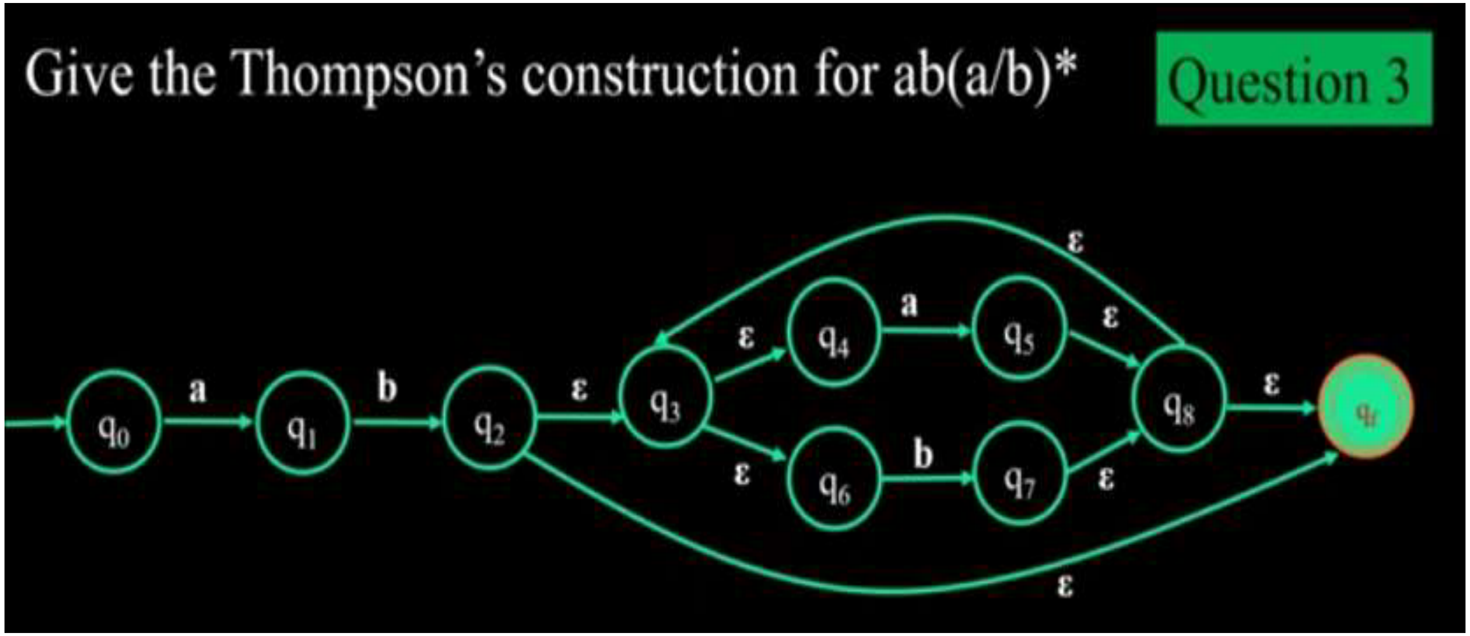
Regular Expression to NFA



Regular Expression to NFA



Regular Expression to NFA



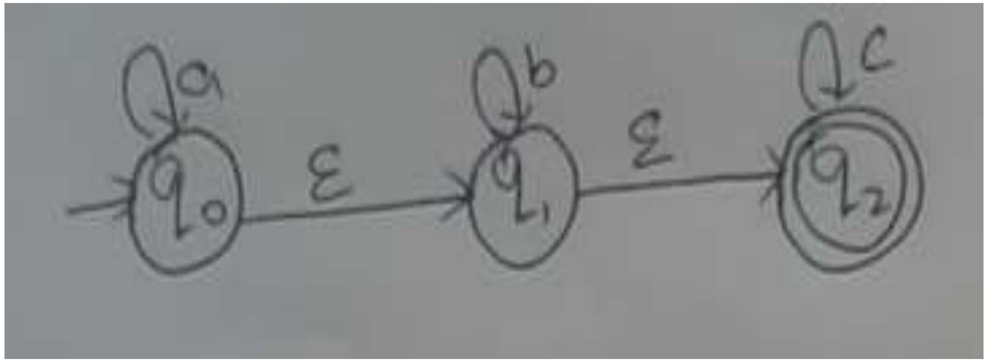
NFA to DFA(Subset Construction)

Converting the NFA into a DFA A Deterministic Finite Automaton (DFA) has at most one edge from each state for a given symbol and is a suitable basis for a transition table. We need to eliminate the ε-transitions by subset construction. Definitions Consider a single state s. Consider a set of states T

Operation	Description
ε-closure(s)	Set of NFA states reachable from NFA state **s** on ε-transitions alone
ε-closure(T)	Set of NFA states reachable from set of states **T** on ε-transitions alone
move(T,a)	Set of states to which there is a transition on input symbol **a** from some NFA state in **T**

3. ε-NFA to DFA

- Convert ε-NFA to DFA for the regular expression a*b*c*
- ***Solution:***
- Step 1: construct the epsilon NFA



- Step 2:
  - Find the ε-closure (q0) = x = A
  - δ(A,a) = q → ε-closure(q)
  - δ(A,b) = r → ε-closure (r)
  - δ(A,c) = s → ε-closure(s)

NFA to DFA(Subset Construction)

The start state of Equivalent DFA is Episilon Closure(0)

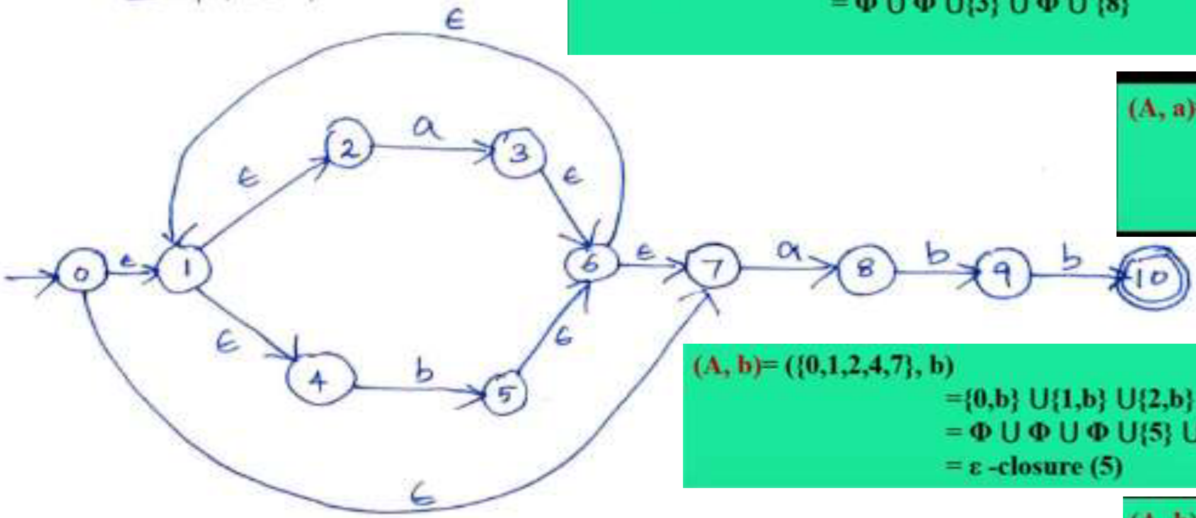
Convert ε-NFA to DFA.

(a|b)* abb ← RE.

1. RE to ε-NFA.

(i) a|b

(ii) (a|b)*



Start with the start state:

ε-closure(0):{0,1,2,4,7} = A

(A, a)= ({0,1,2,4,7}, a) = {0,a} U{1,a} U{2,a} U{4,a} U{7,a}
= Φ U Φ U{3} U Φ U {8}

(A, a)= ({0,1,2,4,7}, a) = {0,a} U{1,a} U{2,a} U{4,a} U{7,a}
= Φ U Φ U{3} U Φ U {8}
= ε -closure (3) U ε -closure (8)
= {1,2,3,4,6,7,8}=B

(A, b)= ({0,1,2,4,7}, b)
={0,b} U{1,b} U{2,b} U{4,b} U{7,b}
= Φ U Φ U Φ U{5} U Φ
= ε -closure (5)

(A, b)= ({0,1,2,4,7}, b)
={0,b} U{1,b} U{2,b} U{4,b} U{7,b}
= Φ U Φ U Φ U{5} U Φ
= ε -closure (5)= {1,2,4,5,6,7}=C

i) ε-closure(0) = {0, 1, 2, 4, 7} = A

ii) A $\xrightarrow{a}$ {3, 8} → ε-closure (3,8) = {1, 2, 3, 4, 6, 7, 8} = B

A $\xrightarrow{b}$ {5} → ε-closure (5) = {1, 2, 4, 5, 6, 7} = C

NFA to DFA(Subset Construction)

iii) $B \xrightarrow{a} \{3,8\} \rightarrow \epsilon\text{-closure}(3,8) = B$

$B \xrightarrow{b} \{5,9\} \rightarrow \epsilon\text{-closure}(5,9) = \{1,2,4,5,6,7,9\} = D$

iv) $C \xrightarrow{a} \{3,8\} \rightarrow \epsilon\text{-closure}(3,8) = B$

$C \xrightarrow{b} \{5\} \rightarrow \epsilon\text{-closure}(5) = C$

v) $D \xrightarrow{a} \{3,8\} \rightarrow \epsilon\text{-closure}(3,8) = B$

$D \xrightarrow{b} \{5,10\} \rightarrow \epsilon\text{-closure}(5,10) = \{1,2,4,5,6,7,10\} = E$

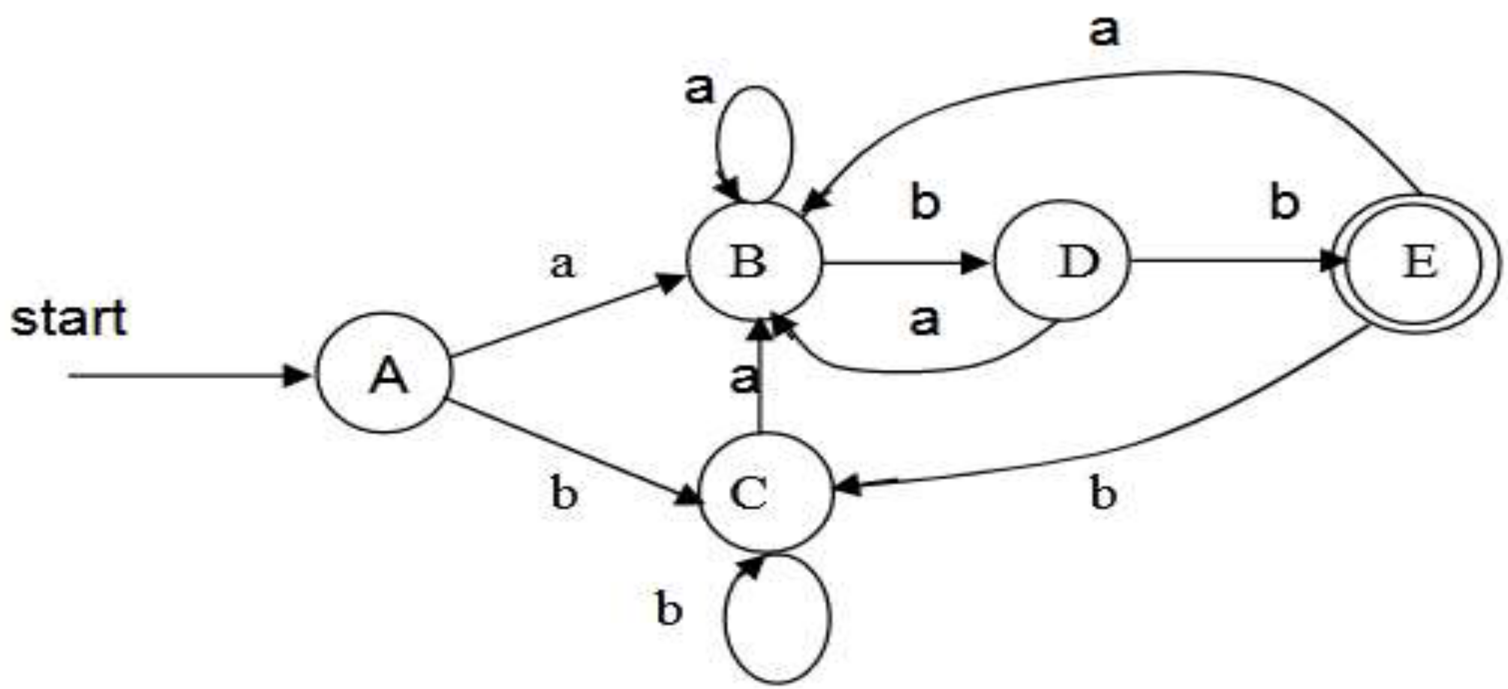
vi) $E \xrightarrow{a} \{3,8\} \rightarrow B$

$E \xrightarrow{b} \{5\} \rightarrow C$

		a	b
→	A	B	C
	B	B	D
	C	B	C
	D	B	E
*	E	B	C

← DFA.

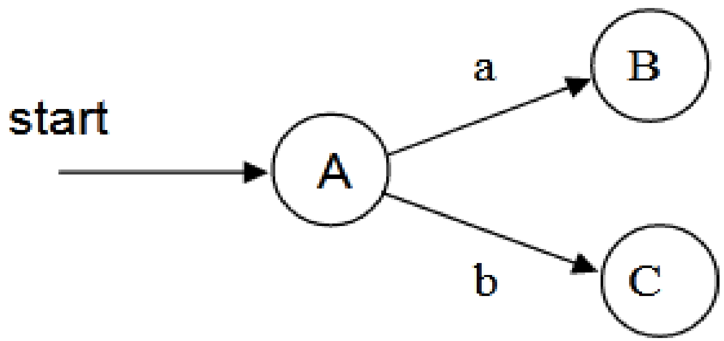
NFA to DFA(Subset Construction)



NFA to DFA

Start the Conversion 1. Begin with the start state 0 and calculate ε-closure(0). a. the set of states reachable by ε-transitions which includes 0 itself is { 0,1,2,4,7}. This defines a new state A in the DFA   A = {0,1,2,4,7}

.We must now find the states that A connects to. There are two symbols in the language (a, b) so in the DFA we expect only two edges: from A on a and from A on b. Call these states B and C:



NFA to DFA

We find B and C in the following way: Find the state B that has an edge on a from A

a. start with A{0,1,2,4,7}. Find which states in A have states reachable by a transitions. This set is called move(A,a) The set is {3,8}:
   move(A,a) = {3,8}

b. now do an ε-closure on move(A,a). Find all the states in move(A,a) which are reachable with ε-transitions. We have 3 and 8 to consider. Starting with 3 we can get to 3 and 6 and from 6 to 1 and 7, and from 1 to 2 and 4. Starting with 8 we can get to 8 only. So the complete set is {1,2,3,4,6,7,8}.
So ε-closure(move(A,a)) = B = {1,2,3,4,6,7,8}

This defines the new state B that has an edge on a from A

NFA to DFA

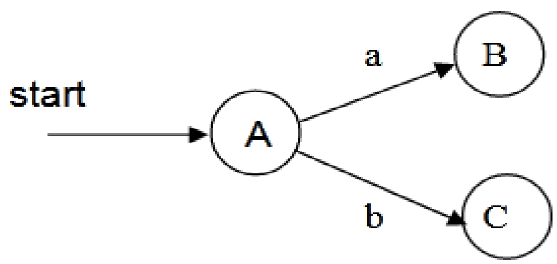
Find the state C that has an edge on b from A

c. start with A{0,1,2,4,7}. Find which states in A have states reachable by b transitions. This set is called move(A,b) The set is {5}:

move(A,b) = {5}

d. now do an ε-closure on move(A,b). Find all the states in move(A,b) which are reachable with ε-transitions. We have only state 5 to consider. From 5 we can get to 5, 6, 7, 1, 2, 4. So the complete set is {1,2,4,5,6,7}. So ε-closure(move(A,a)) = C = {1,2,4,5,6,7}

This defines the new state C that has an edge on b from A



A={0,1,2,4,7}
B={1,2,3,4,6,7,8}
C={1,2,4,5,6,7}

Now that we have B and C we can move on to find the states that have a and b transitions from B and C.

NFA to DFA

Find the state that has an edge on a from B

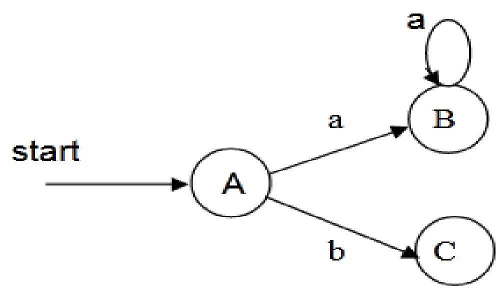
e. start with B{1,2,3,4,6,7,8}. Find which states in B have states reachable by atransitions. This set is called move(B,a) The set is {3,8}:

move(B,a) = {3,8}

f. now do an ε-closure on move(B,a). Find all the states in move(B,a) which are reachable with ε-transitions. We have 3 and 8 to consider. Starting with 3 we can get to 3 and 6 and from 6 to 1 and 7, and from 1 to 2 and 4. Starting with 8 we can get to 8 only. So the complete set is {1,2,3,4,6,7,8}.

So ε-closure(move(A,a)) = {1,2,3,4,6,7,8}

which is the same as the state B itself. In other words, we have a repeating edge to B:



A={0,1,2,4,7}
B={1,2,3,4,6,7,8}
C={1,2,4,5,6,7}

NFA to DFA

Find the state D that has an edge on b from B

g. start with B{1,2,3,4,6,7,8}. Find which states in B have states reachable by b transitions. This set is called move(B,b) The set is {5,9}:

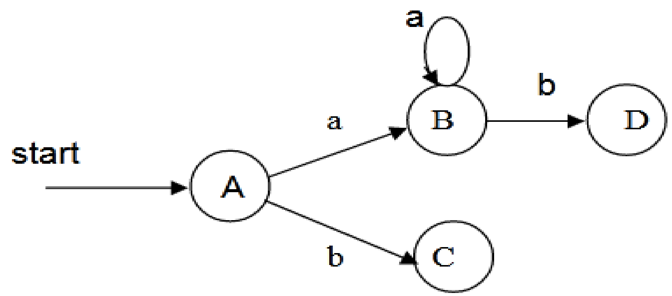
move(B,b) = {5,9}

h. now do an ε-closure on move(B,b). Find all the states in move(B,b) which are reachable with ε-transitions. From 5 we can get to 5, 6, 7, 1, 2, 4. From 9 we get to 9 itself. So the complete set is {1,2,4,5,6,7,9}}.

So ε-closure(move(B,a)) = D = {1,2,4,5,6,7,9}

This defines the new state D that has an edge on b from B

A={0,1,2,4,7}, B={1,2,3,4,6,7,8}, C={1,2,4,5,6,7}, D{1,2,4,5,6,7,9}



NFA to DFA

Find the state that has an edge on a from D

i. start with D{1,2,4,5,6,7,9}. Find which states in D have states reachable by a transitions. This set is called move(D,a) The set is {3,8}:

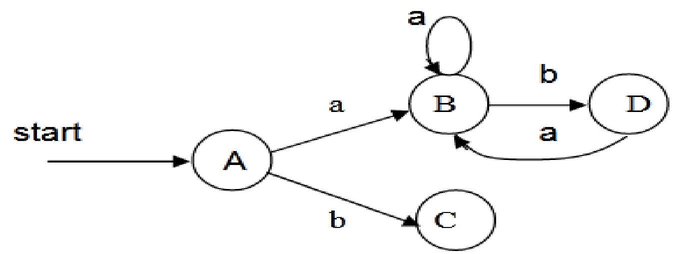
move(D,a) = {3,8}

j. now do an ε-closure on move(D,a). Find all the states in move(B,a) which are reachable with ε-transitions. We have 3 and 8 to consider. Starting with 3 we can get to 3 and 6 and from 6 to 1 and 7, and from 1 to 2 and 4. Starting with 8 we can get to 8 only. So the complete set is {1,2,3,4,6,7,8}. So

ε-closure(move(D,a)) = {1,2,3,4,6,7,8} =B

This is a return edge to B:

A={0,1,2,4,7}, B={1,2,3,4,6,7,8}, C={1,2,4,5,6,7}, D{1,2,4,5,6,7,9}



NFA to DFA

Find the state E that has an edge on b from D

k. start with D{1,2,4,5,6,7,9}. Find which states in D have states reachable by btransitions. This set is called move(B,b) The set is {5,10}:

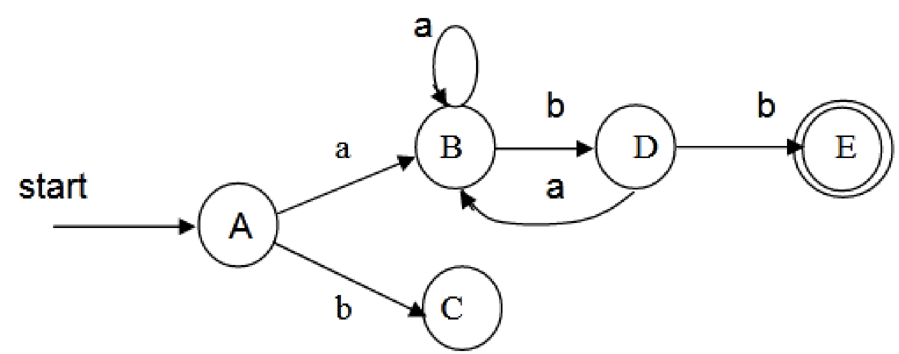
move(D,b) = {5,10}

l. now do an ε-closure on move(D,b). Find all the states in move(D,b) which are reachable with ε-transitions. From 5 we can get to 5, 6, 7, 1, 2, 4. From 10 we get to 10 itself. So the complete set is {1,2,4,5,6,7,10}}. So ε-closure(move(D,b) = E = {1,2,4,5,6,7,10}

This defines the new state E that has an edge on b from D.

Since it contains an accepting state, it is also an accepting state.

A={0,1,2,4,7}, B={1,2,3,4,6,7,8}, C={1,2,4,5,6,7}, D={1,2,4,5,6,7,9}, E={1,2,4,5,6,7,10}



NFA to DFA

We should now examine state C

Find the state that has an edge on a from C

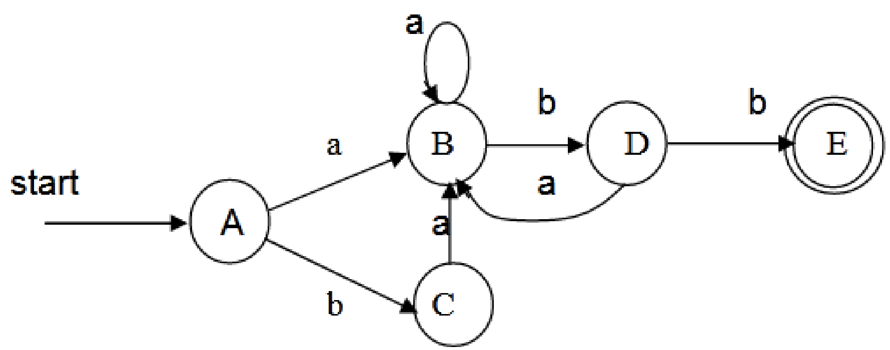
m. start with C{1,2,4,5,6,7}. Find which states in C have states reachable by atransitions.

This set is called move(C,a) The set is {3,8}:

move(C,a) = {3,8}

we have seen this before. It's the state B

A={0,1,2,4,7}, B={1,2,3,4,6,7,8}, C={1,2,4,5,6,7}, D={1,2,4,5,6,7,9}, E={1,2,4,5,6,7,10}



NFA to DFA

Find the state that has an edge on b from C

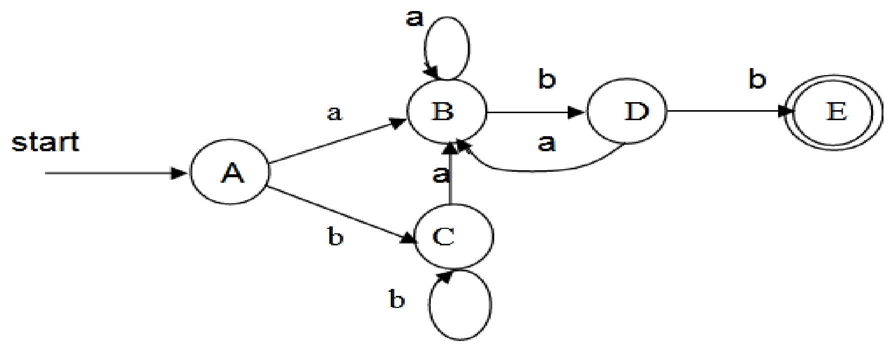
n. start with C{1,2,4,5,6,7}. Find which states in C have states reachable by btransitions. This set is called move(C,b) The set is {5}:

move(C,b) = {5}

o. now do an ε-closure on move(C,b). Find all the states in move(C,b) which are reachable with ε-transitions. From 5 we can get to 5,6,7,1,2,4. which is C itself So

ε-closure(move(C,b)) = C

This defines a loop on C



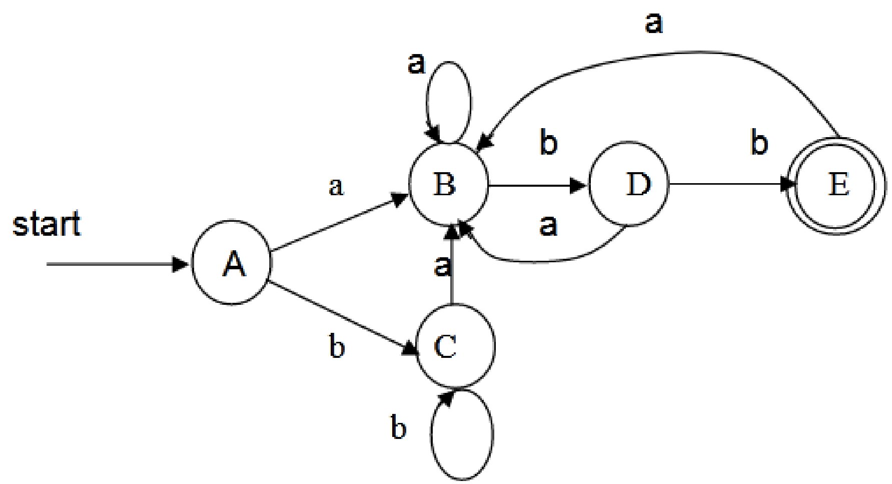
NFA to DFA

Find the state that has an edge on a from E

p. start with E{1,2,4,5,6,7,10}. Find which states in E have states reachable by a transitions. This set is called move(E,a) The set is {3,8}:

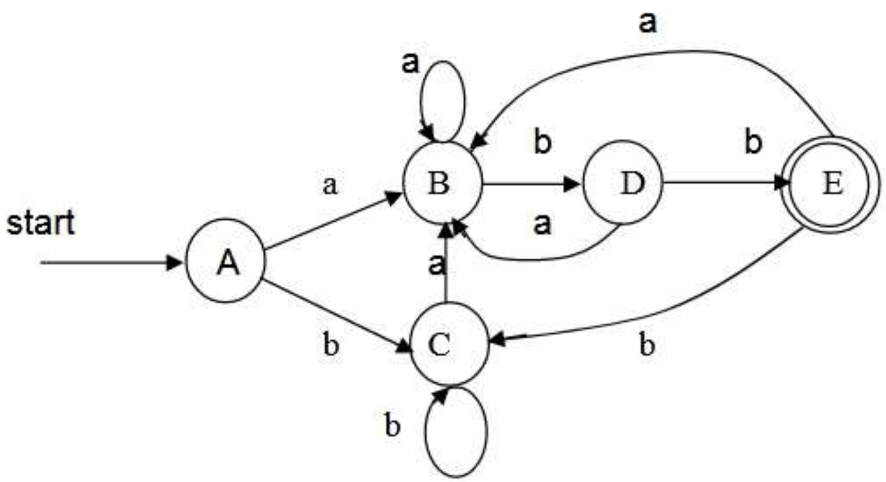
move(E,a) = {3,8}

We saw this before, it's B. So



NFA to DFA

Find the state that has an edge on b from E
q. start with E{1,2,4,5,6,7,10}. Find which states in E have states reachable by btransitions. This set is called move(E,b) The set is {5}:
move(A,b) = {5} We've seen this before.
It's C. Finally



NFA to DFA

That’s it ! There is only one edge from each state for a given input character. It’s a DFA. Disregard the fact that each of these states is actually a group of NFA states. We can regard them as single states in the DFA. In fact it also requires other as an edge beyond E leading to the ultimate accepting state. Also the DFA is not yet optimized (there can be less states).

However, we can make the transition table so far. Here it is:

State	Input **a**	Input **b**
A	B	C
B	B	D
C	B	C
D	B	E
E	B	C


Thank you